INTERNATIONAL JOURNAL FOR NUMERICAL AND ANALYTICAL METHODS IN GEOMECHANICS
Int. J. Numer. Anal. Meth. Geomech., **22**, 777–790 (1998)

ANALYSIS OF PILED RAFTS

I. M. SMITH[1,*] AND A. WANG[2]

[1]*University of Manchester, School of Engineering, Manchester, M13 9PL, U.K.*
[2]*National University of Singapore, Department of Civil Engineering, Singapore*

SUMMARY

Analysis of piled raft foundations, taking account of their full three-dimensional complexity, can be accomplished by modern finite element analysis techniques. The characteristics of the (preconditioned conjugate gradient) numerical method applied to this problem are analysed, and then the method is used in a field problem of a raft subjected to very rapidly varying loading patterns. © 1998 John Wiley & Sons, Ltd.

Key words: piles; rafts; finite elements; preconditioning; conjugate gradients

1. INTRODUCTION

Clancy and Randolph[1] and Griffiths *et al.*[2] have described approximate analysis procedures for piled rafts based on the work of Chow.[3] These 'hybrid' finite element techniques are assumed to be necessary due to the excessive storage requirements for traditional finite element discretizations of the full three-dimensional problem. Modern approaches, which seek to utilize the power of vector and parallel computers, e.g. Smith and Griffiths,[4] have essentially removed the limitations based on storage of assembled matrices. The success of these (iterative) solution techniques then depends only on the speed of convergence of the equation solution process and on the efficiency with which parallelism can be exploited. It will be shown that very large three-dimensional discretisations of piled rafts can be solved in a matter of minutes on a parallel computer. While still not 'routine', such analyses can be justified for sensitive problems.

2. PCG STATISTICS

Iterative techniques for equation solution such as preconditioned conjugate gradients (PCG), minimum residual (MINRES) and generalized minimum residual (GMRES) have been intensively re-examined over the past ten years or so. Their essential feature is that they permit 'element-by-element' or 'mesh-free' strategies which avoid the need to assemble large global matrices such as the finite element 'stiffness' matrix.[5,6] Furthermore, they are highly vectorizable and parallelizable and can make very efficient use of modern computers which achieve their performance gains by these methods.[7]

*Correspondence to: I. M. Smith, School of Engineering, University of Manchester, Manchester, M13 9PL, UK

CCC 0363–9061/98/100777–14$17.50

Received 17 July 1997
Revised 30 October 1997

Table I. PCG algorithm in Fortran 90

```
! ------- IF X =.0 P AND R ARE JUST LOADS BUT IN GENERAL P=R=LOADS-A*X -------
! -------------------------- SO FORM R = A * X -------------------------------------------------------
        R = .0  ;  X(0) = .0
        ELEMENTS_2A  :  DO IEL = 1, NUMBER_OF_ELEMENTS
                                G_PMUL (:,IEL)  =  X(G_G (:,IEL))     ! GATHER
        END DO ELEMENTS_2A
        G_UTEMP = MATMUL (KM,G_PMUL)
          ELEMENTS_2B: DO IEL = 1 , NUMBER_OF_ELEMENTS
                DO I = 1 , NDOF
                   R (G_G(I,IEL)) = R(G_G(I,IEL)) + G_UTEMP(I,IEL)     ! SCATTER
                END DO
        END DO ELEMENTS_2B    ; U(0) = .0
! ------------------------------- NOW PRECONDITION R AND P ------------------------------------------

R(0) = .0  ;  R = LOADS - R ; D = DIAG_PRECON * R  ;  P = D
! ----------------- SOLVE THE SIMULTANEOUS EQUATIONS BY PCG ---------------------------------
          CJITERS = 0
        CONJUGATE_GRADIENTS:  DO
          CJITERS = CJITERS + 1;          U = .0
        ELEMENTS_3 : DO IEL = 1,  NUMBER_OF_ELEMENTS
                G_PMUL (:,IEL) = P(G_G(:,IEL))     ! GATHER
        END DO ELEMENTS_3
        G_UTEMP = MATMUL (KM,G_PMUL)
          ELEMENTS_3A: DO IEL = 1, NUMBER_OF_ELEMENTS
                        DO I = 1 , NDOF
                           U((G_G(I,IEL)) = U(G_G(I,IEL)) + G_UTEMP (I,IEL)     ! SCATTER
                        END DO
        END DO ELEMENTS_3A  ;  U(0) = .0
! ----------------------------------------------- PCG PROCESS -------------------------------------------------
        UP =DOT_PRODUCT(R,D); ALPHA=UP/DOT_PRODUCT(P,U)
        XNEW = X + P* ALPHA; R = R - U*ALPHA; R(0) = .0 ; D = DIAG_PRECON*R
        BETA = DOT_PRODUCT(R,D)/UP; P = D + P * BETA
        BIG = .0; CJ_CONVERGED = .TRUE.
        DO I = 1, NEQ; IF(ABS(XNEW(I))>BIG)BIG=ABS(XNEW(I)); END DO
        DO I = 1, NEQ; IF(ABS (XNEW(I)-X(I))/BIG>CJTOL)CJ_CONVERGED=.FALSE.;END DO
        X = XNEW
        IF(CJ_CONVERGED.OR.CJITERS==CJITS) EXIT
           END DO CONJUGATE_GRADIENTS
!---------------------------------------------END OF PCG PROCESS --------------------------------------------
```


For completeness, the PCG algorithm, in Fortran 90,[8] is given in Table I.
Points to notice are:

1. Solution is of $A^*X = \text{LOADS}$.
2. Simple diagonal preconditioning DIAG_PRECON has been used.
3. In non-linear applications where 'loads' are applied step by step, it is essential to use the solution at the end of the previous step as the first guess on the current step. Typically this accelerates convergence by a factor of 2.
4. A global index matrix G_G facilitates the gather and scatter operations.
5. The element stiffness matrices are KM.

A regular finite element mesh is shown in Figure 1 with an embedded 'raft' element at one corner of a foundation block of 'soil'. If the soil, represented by elastic elements, were uniform, the stiffnesses of all soil elements would be identical and it would only be necessary to calculate and store one of them. Thus, there would only be two distinct element stiffnesses, one for raft and the other for soil. Although this is a very simplified case, it should be obvious that in many situations, involving, for example, layered deposits there will be 'groups' of elements with the same stiffness properties. In non-linear calculations, if 'constant stiffness' iterations are performed, the identities of these groups will be preserved throughout the calculation. The sensitivity of the PCG process to the initial guess of X is illustrated by Figure 2. The convergence of the PCG process is dependent on the eigenvalue spectrum of the A matrix. For foundation analyses, a typical demonstration of this effect is shown in Figure 3, where, as the 'soil' Poisson's ratio is increased towards 0·5, iterations to convergence increase, by a factor of about 4 in this case.

The solutions leading to Figure 3 involved 125 elements and about 1400 degrees of freedom. For the lower values of Poisson's ratio of the foundation, iterations to convergence/number of equations (degrees of freedom) was about 0·06. Fortunately, as problem sizes increase, this figure drops dramatically as shown in Figure 4. When there are 10,000 elements the number of iterations per degree of freedom has dropped to 0·001 and for one million equations it is about 0·0002. In practical terms, very large systems of equations involve a few hundred iterations of the preconditioned conjugate gradient algorithm on the first solution, and about half that number on subsequent re-solutions.

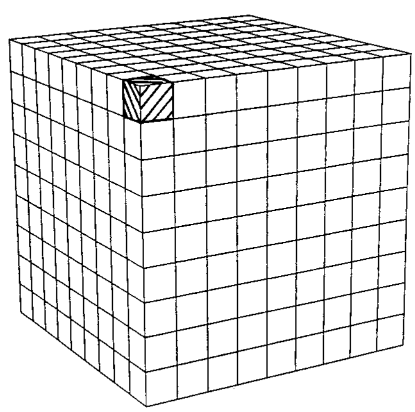


Figure 1. Basic raft/soil idealization

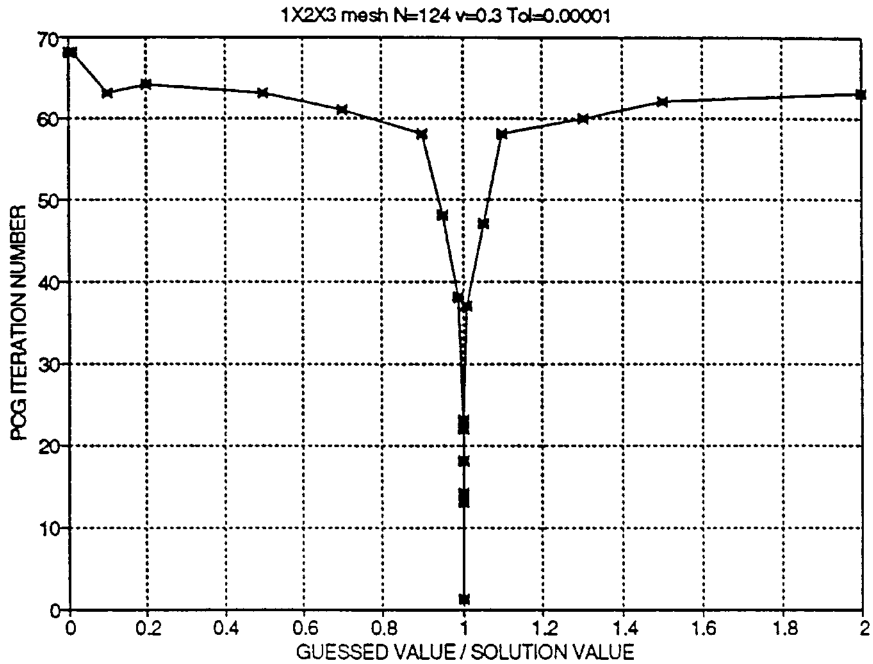


Figure 2. Effect of initial guess on PCG iterations

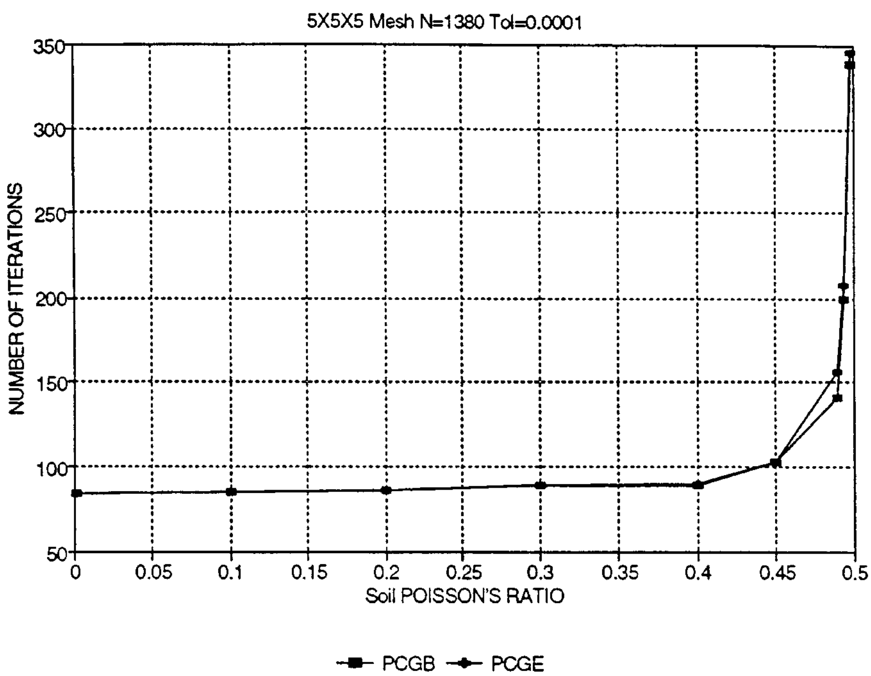


Figure 3. Effect of soil Poisson's ratio on PCG iterations

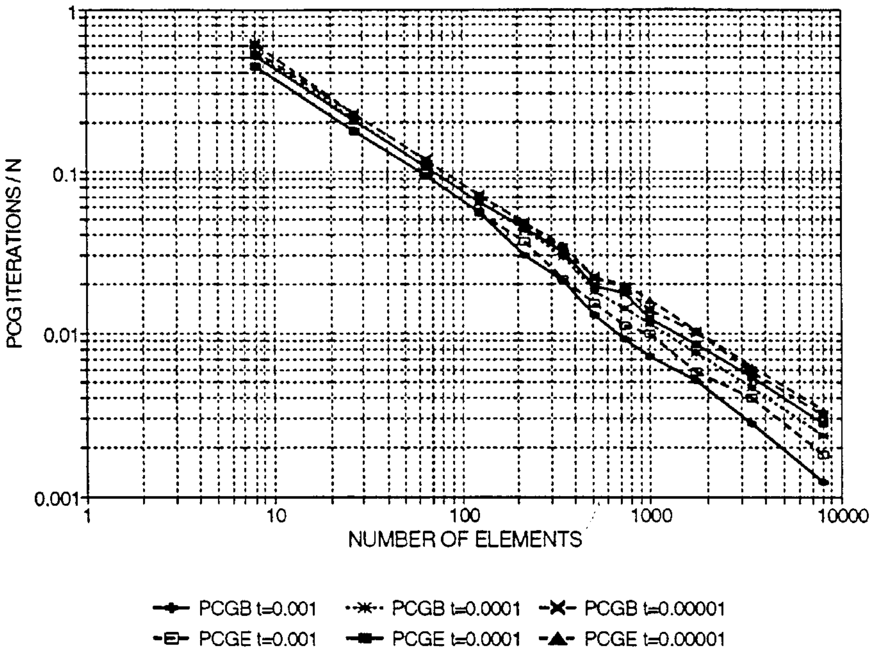


Figure 4. PCG iterations per freedom with increasing problem size

3. PARALLEL PROCESSING

The algorithm shown in Table I is potentially highly parallelizable. It can be seen to be characterized by two main sections making up the loop named CONJUGATE_GRADIENTS. First there is a 'gather' operation, illustrated by the loop named ELEMENTS_3, followed by a matrix multiply (MATMUL) and a 'scatter' operation, illustrated by the loop named ELEMENTS_3A.

As shown in Table II, a typical element stiffness matrix KM, or a block of them, may be associated with non-contiguous parts of a vector P, grouped together as G_PMUL, in order for MATMUL to be carried out.

In the subsequent 'vector' operations, beginning with UP=DOT_PRODUCT (R, D) the natural chop of X,R,D, etc. over the processors is as shown in the lower part of Table II. But, for the gather–multiply–scatter section in the top part of Table II communication between processors is necessary to ensure that the appropriate parts of G_PMUL are available to be multiplied by KM (or by blocks of KMs if preferred). The efficiency with which this communication can be handled by a given parallel architecture is the key to the success of this type of algorithm.[7] It has been tested on the following systems:

Cray T3D, T3E
Cray J90
IBM SP2
Fujitsu VPP
Silicon Graphics Origin 2000
Silicon Graphics Workstation Cluster

Table II. Incompatible chopped section of NEQ equations

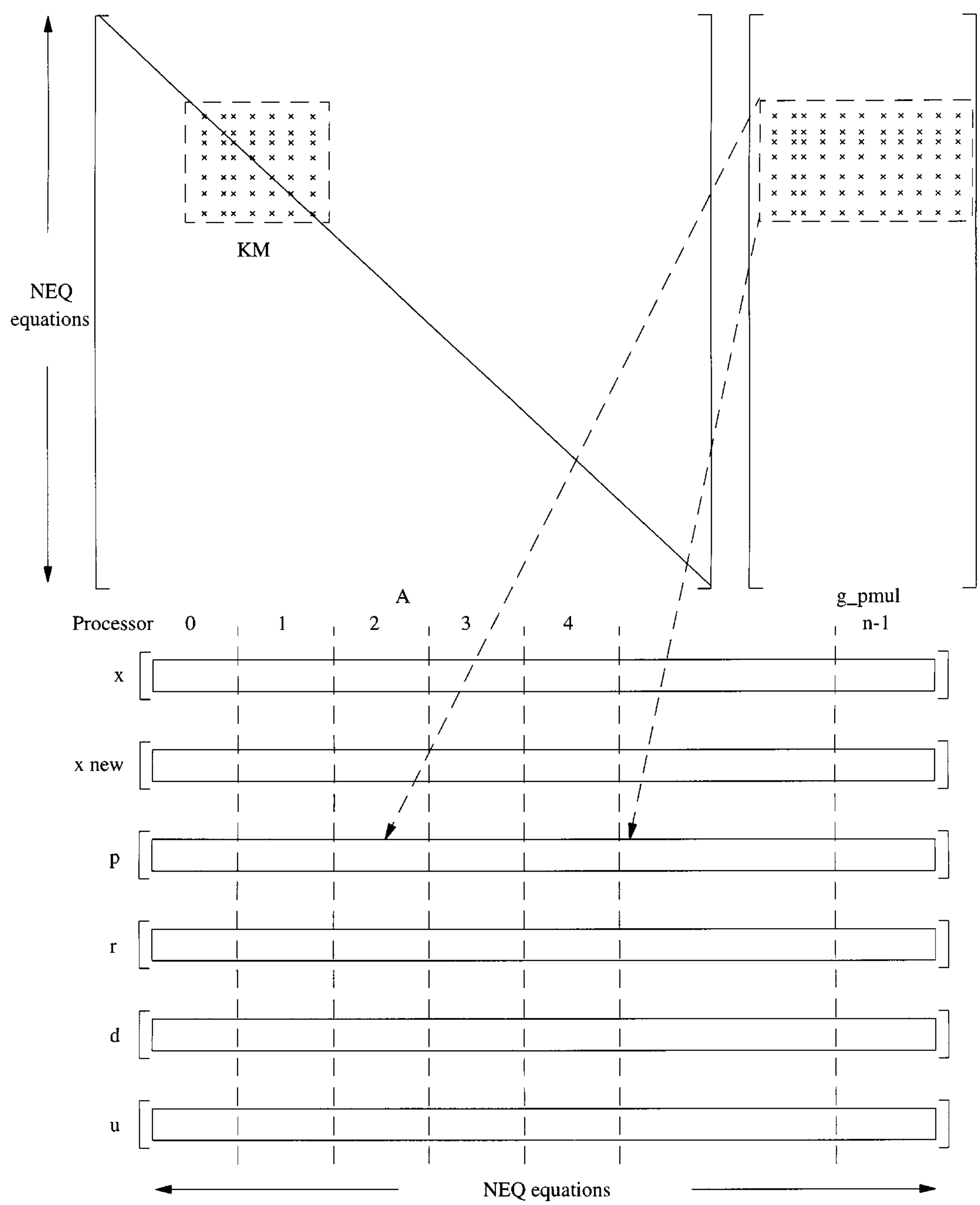


Table III. Parallelisation statistics, Cray T3D, MPI (partial analysis)

		Serial EL98	8	16	32	64	128	256	512
Total	Wall	3584	493	257	141	80	52	50·1	52.0
Bcast.	Wall		8·1	8·3	9·1	10·0	11·7	18·1	24·6
Local g	Wall		6·4	2·6	1·9	1·7	3·4	11·4	15·8
Load inc.	Wall		478	246	130	68	37	21	11·9
	Spdup		1·0	1·9	3·7	7·0	12·9	22·8	40·2
	(max)			(2)	(4)	(8)	(16)	(32)	(64)
PCG	Wall		459	236	125	66	36	20	11·5
Solver::	Mflp/s		504	979	1852	3505	6446	11 550	20 110
	%Peak		42%	41%	39%	37%	34%	30%	26%
Peak	Mflp/s	133	1200	2400	4800	9600	19 200	38 400	76 800

using MPI. This ensures portability and typical results for the Cray T3D implementation are shown in Table III. The problem is of the piled raft foundation type with about one million degrees of freedom (NEQ $\simeq$ 1,000,000). Points to notice are that the complete solution procedure took about 8 min when up to 512 processors were used. The minimum number of processors on which the job could be run was eight and the speed of computation was about 500 Mflop/s or a rather disappointing 42 per cent of peak performance of this hardware. However, as the number of processors (communication) increased, the degradation in speed was acceptable, 26 per cent of peak being achieved with 512 processors. The heavy computation was running at 20 Gflop/s, a highly creditable performance. A similar pattern of performance has been obtained on all the systems listed above. From the point of view of civil engineering applications, the performance on 12 Silicon Graphics workstations using ATT communication is particularly encouraging since hardware of this type could be made available in practice at affordable cost.

4. PROGRAM VALIDATION

Figure 5 shows a small model of a quarter of a 5×5 piled raft. Elastic behaviour of piles, raft and soil is assumed. The following notation is used:

Soil		Pile		Raft	
Young's modulus	E_s	Young's modulus	E_p	Young's modulus	E_r
Poisson's ratio	v_s	Length	L	Poisson's ratio	v_r
		Diameter	d	Thickness	t
		Spacing	s	Length	L

Results are given in terms of the relative stiffness of raft compared to soil,

$$K_r = \frac{E_r(l - v_s^2)}{E_s}\left(\frac{t}{L}\right)^3$$

and of pile relative to soil E_p/E_s.

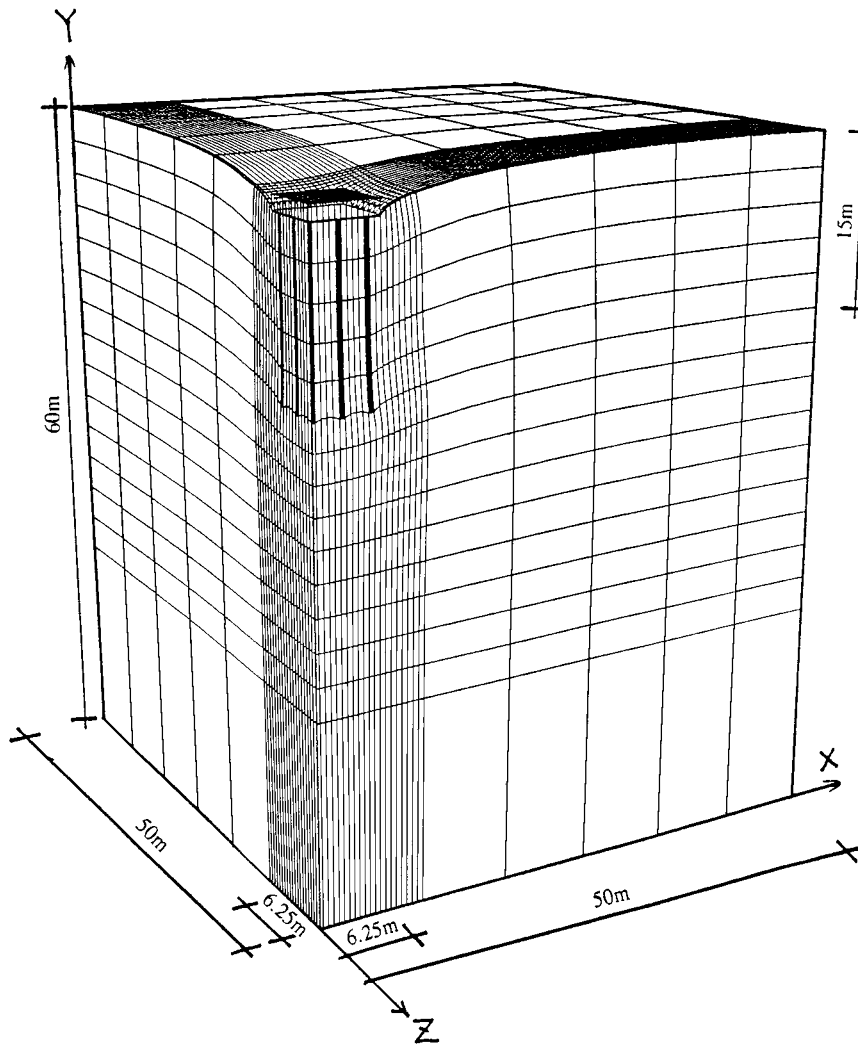


Figure 5. Displaced mesh for piled raft

Figure 6 shows the pile load distribution as the raft varies in stiffness from almost completely flexible, $K_r = 0{\cdot}01$ to almost completely rigid, $K_r = 1000$. For the flexible raft, load is uniformly distributed amongst the piles whereas for a very stiff raft, the central pile carries 30 per cent of the average pile load and the corner piles 200 per cent in good agreement with results long established by Poulos and Davis.[9] The range over which raft flexibility plays a significant role can be seen to be $0{\cdot}01 \leqslant K_r \leqslant 1$ for the chosen values of K_p, L/d and s/d. Similar conclusions can be reached for decreasing values of K_p (see Figure 7).

The load carried by the piles as a proportion of the total applied load is shown in Figure 8. It can be seen that the raft stiffness does not affect the load carried by the piles, only its distribution. Even for very stiff piles, the raft carries some 30 per cent of the applied load.

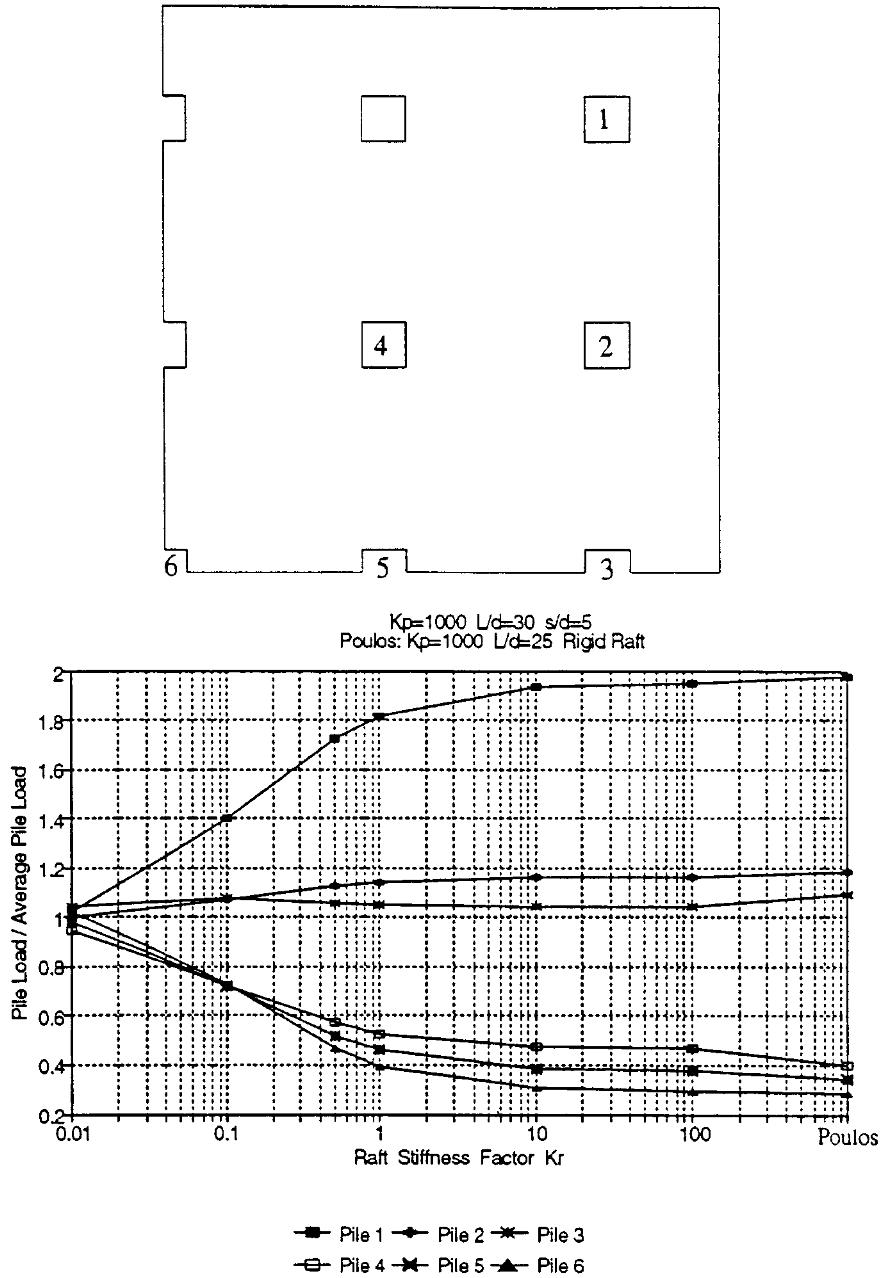
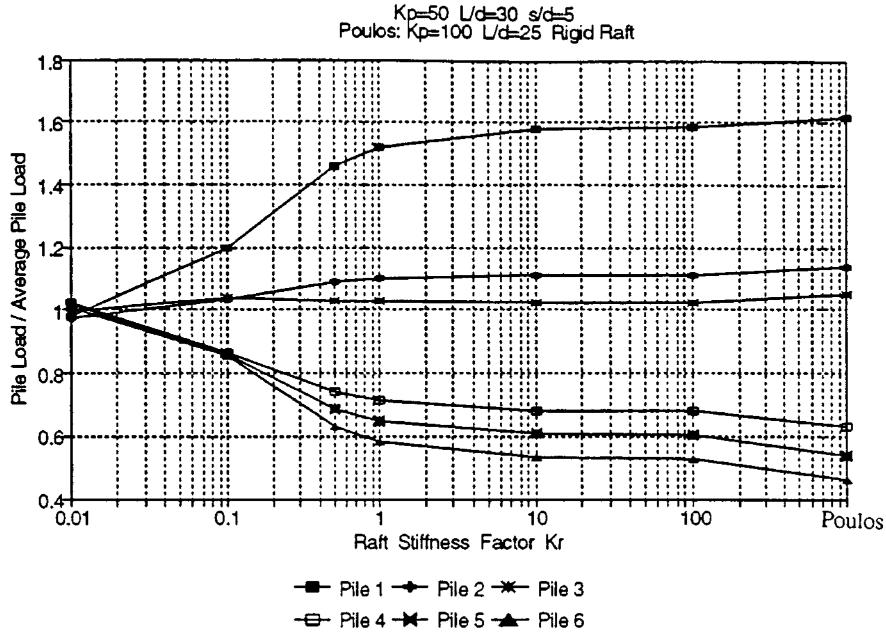


Figure 6. Pile load distribution for rafts of varying stiffness

5. FIELD EXAMPLE

When loads of very different magnitudes are to be supported in close proximity, usual design practice is to provide construction joints between the heavily and lightly loaded parts of the



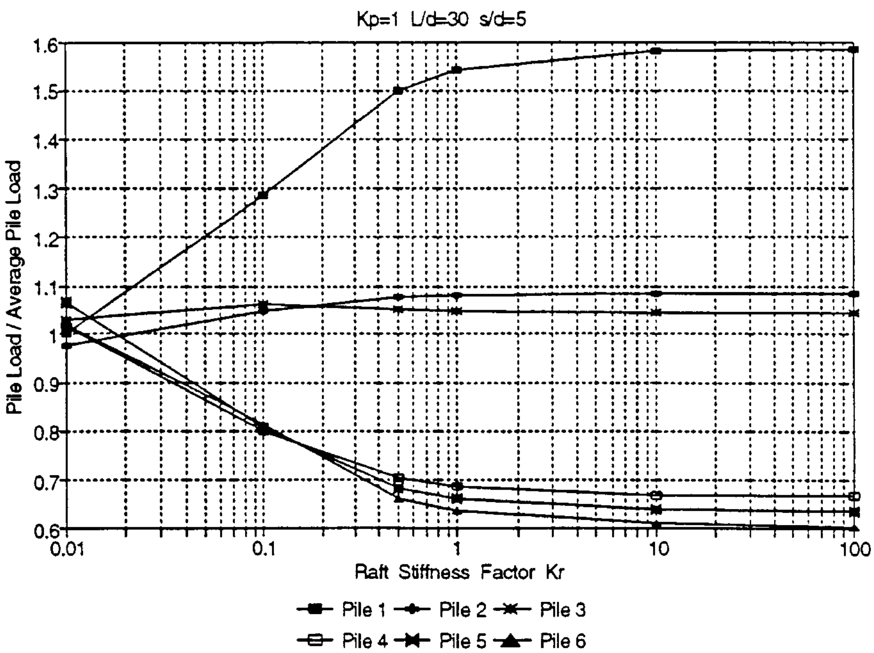


Figure 7. Pile load distributions for piles of varying stiffness

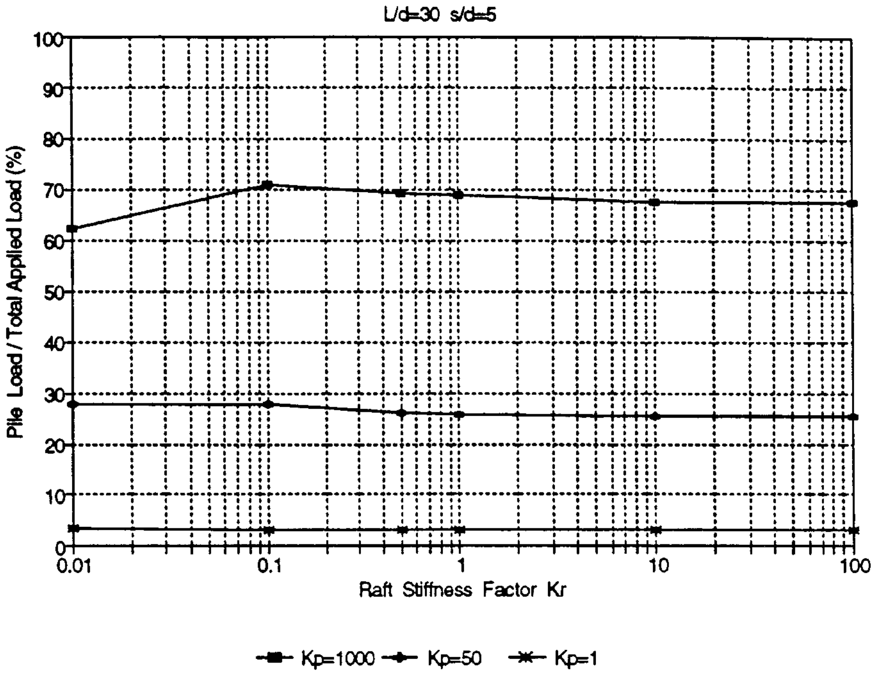


Figure 8. Total pile loads for varying pile and raft stiffness

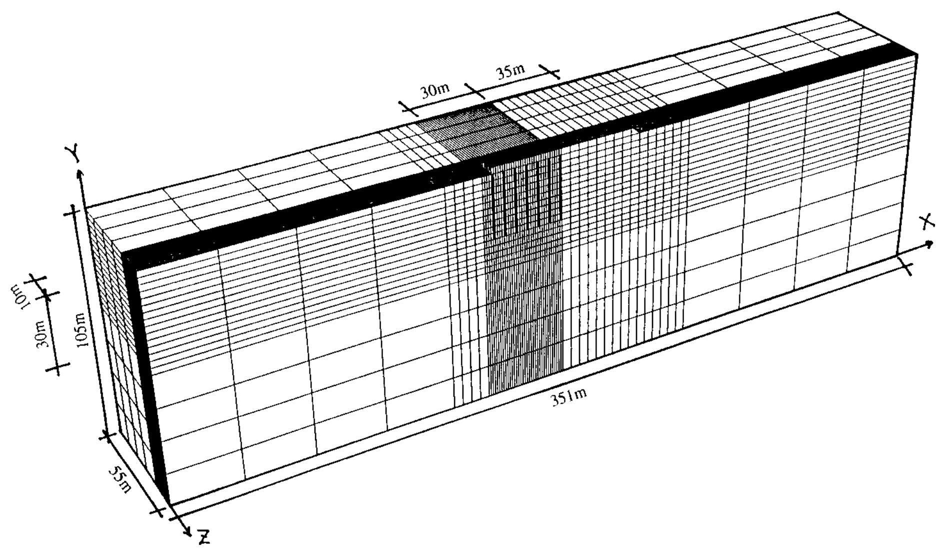


Figure 9. Differentially loaded piled raft

foundation. For some sensitive structures this is unacceptable, and integrity of a complete raft subjected to very unevenly distributed loads must be preserved.

Such an example is shown in Figure 9. A raft with plan dimensions 65×20 m and a thickness of 1·5 m carries 400 kN/m^2 over a 30×20 m area (left) and only 100 kN/m^2 over the remaining

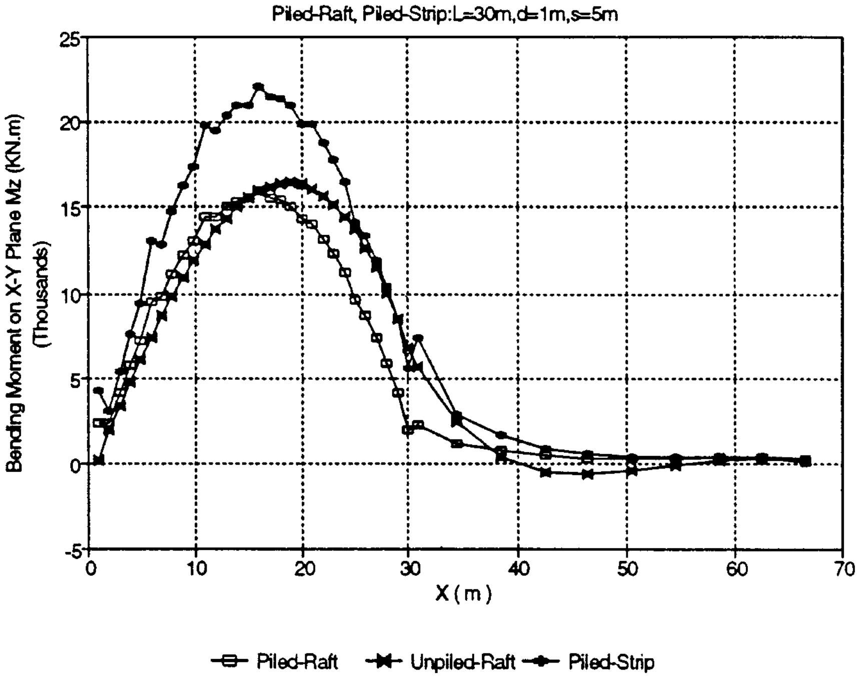


Figure 10. Raft bending moments on x–y plane

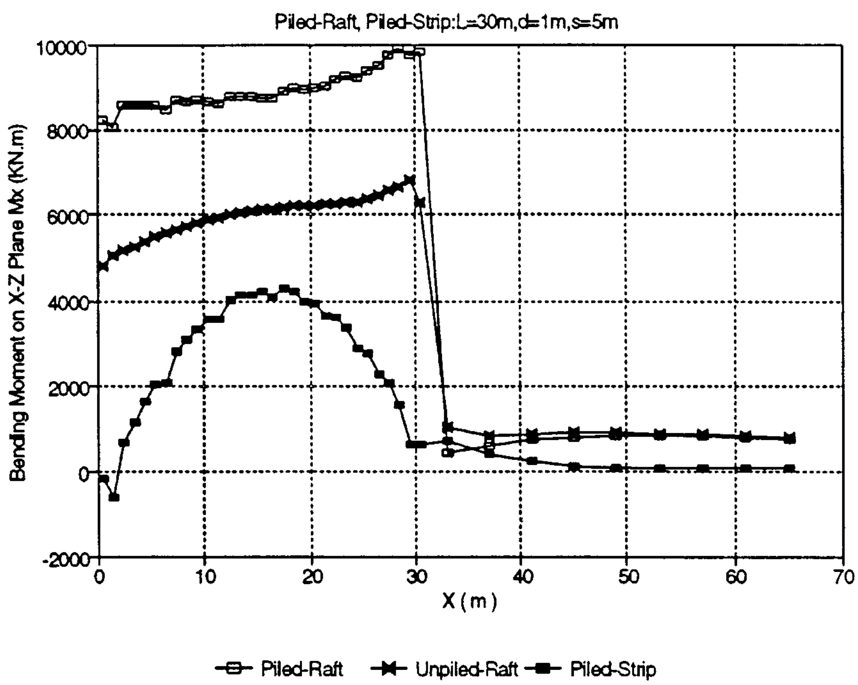


Figure 11. Raft bending moments on x–y plane

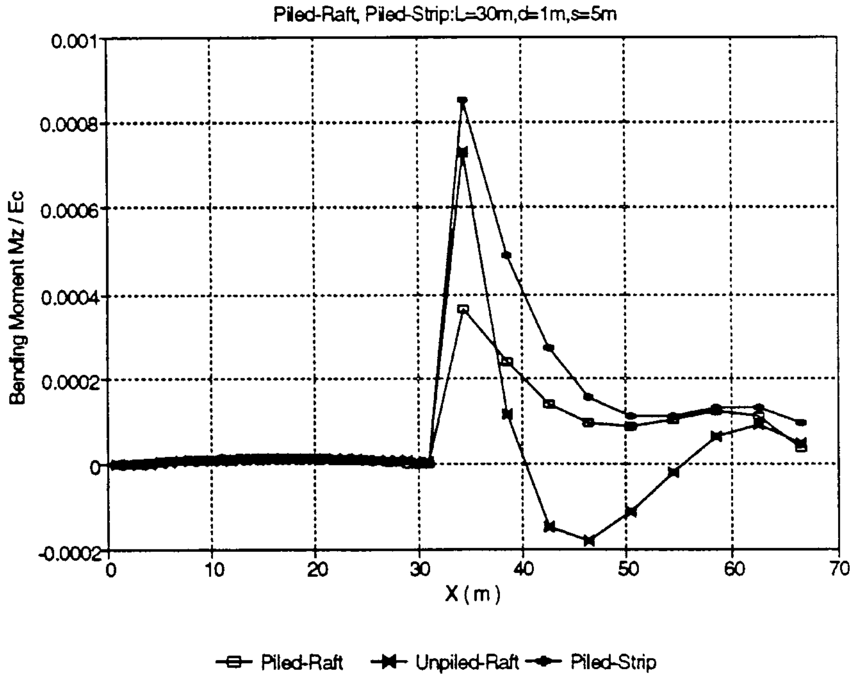


Figure 12. Normalised raft bending moments on x–y plane

35×20 m area (right). The structural design leads to the heavily loaded section consisting of a stiff, heavily reinforced raft while the lightly loaded section can be lightly reinforced.

However, very large bending moments will inevitably be experienced at the junction between the heavily loaded and lightly loaded portions of the raft. In order to obtain a balanced design, it is natural to investigate piling under the heavily loaded section and results are presented here for 1 m diameter piles, 30 m long and at a spacing of 5 m ($L = 30$ m, $d = 1$ m, $s = 5$ m).

Moments along the centreline of the raft on x–z planes are shown in Figure 10 for three analyses: two 3-D analyses, one with piles and one without, and a 2-D strip analysis with piles. It can be seen that the 2-D analysis greatly overestimates the raft bending moments. However, the presence of piles, although it does significantly reduce differential settlements, has only apparently a limited effect on the absolute bending moments, reducing them at the critical junction section by a factor of about 3. The effect of the 3-D analysis is shown in Figure 11 where it can be seen that the 3-D raft carries significant moments in the plane transverse to the long side of the raft although these moments are small relative to the maximum moments in the raft.

What is important is the bending moment distribution relative to the raft section. This is shown in Figure 12 where the moments are normalized by the (reinforced) Young's modulus of the appropriate raft sections. It can be seen that the 3-D piled raft analysis leads to a reduction in relative moment of about 2·5 compared to both the unpiled raft and piled strip analyses.

6. CONCLUSIONS

Using modern methods it is feasible to contemplate analysing sensitive problems in their full three-dimensional complexity. Such analyses have been presented for piled raft foundations. The statistics of preconditioned conjugate gradient (PCG) computations have been presented. These show that quite large problems (around 1 million degrees of freedom) can be solved in a matter of minutes on a supercomputer such as a Cray T3D. More importantly, the methods have been shown to be portable to much cheaper systems, such as clusters of workstations communicating via relatively inexpensive networks. These systems are within the budgets of civil engineering organisations.

A field example of a differentially loaded raft has been analysed, and the benefits of full 3-D analyses confirmed.

7. REFERENCES

1. P. Clancy and M. F. Randolph, 'An approximate analysis procedure for piled raft foundations', *Int. J. Num. Anal. Meth. Geomech.*, **17** (12), 849–869 (1993).
2. D. V. Griffiths, P. Clancy and M. F. Randolph, 'Piled Raft Foundations by finite elements', *Proc. 7th Int. Conf. Computer Methods and Advances in Geomech.*, Cairns, Vol. 2, May 1991, pp. 1153–1157.
3. Y. K. Chow, 'Analysis of vertically loaded pile groups', *Int. J. Numer. Anal. Meth. Geomech.*, **10,** 59–72 (1986).
4. I. M. Smith and D. V. Griffiths, '*Programming the Finite Element Method*', 3rd ed, Wiley, Chichester, 1998, pp. 534.
5. T. J. R. Hughes, R.M. Ferencz and J. O. Hallquist, 'Large-scale vectorised implicit calculations in solid mechanics on a Cray-YMP48 utilising EBE preconditioned conjugate gradients', *Comput. Meth. Appl. Mech. Engng.*, **61,** 215–248 (1987).
6. S. W. Wong, I. M. Smith and I. Gladwell, 'PCG Methods in transient FE analysis, Part II, Second Order Problems', *Int. J. Numer. Meth. Engng.*, **28**(7), pp. 1567–1576 (1989).
7. I. M. Smith and M. A. Pettipher, 'Portable Parallel Computing Environment for General Finite Element Analysis', submitted for publication.
8. I. M. Smith, '*Programming in Fortran 90*', Wiley, Chichester, 1995.
9. H. G. Poulos and E. H. Davis, '*Pile Foundation Analysis and Design*', Wiley, New York, 1980.

